鐵達尼號生存預測

國立臺北商業大學 資訊管理系 二技 二年甲班

10636005 許凱茜　10636019 王欣慈　10636028 李順玉

一、 摘要

如果你搭上了鐵達尼號，有機會生還嗎？鐵達尼號是歷史上非常知名之嚴重的船難事件，我們利用機器學習中的隨機森林模型來預測乘客是否生存。

近年來鐵達尼號的真相，是許多人趨之若鶩想知道的事情，透過機器學習可以讓我們更了解死亡的真相，究竟死亡的人是男是女、是老是年輕、是富是貧，都可以輕易得知。更重要的是我們能夠了解所有乘船因素是否會影響存活機率。

二、 研究背景與目的

傑克與蘿絲的愛情故事感動了無數的人成為了膾炙人口與雋永的故事，也讓我們想更了解鐵達尼號的全貌及當初發生的事，究竟是否如同電影中所呈現的結局，獲救的是否都是女性及孩童，也為了更了解機器學習如何運作，並且能夠透過這次的實作經驗提升我們的專業技能，我們決定挑戰 Kaggle 中知名的題目：鐵達尼號生存預測，我們也很好奇如果我們在那艘船上，以我們自身的條件是否能夠存活呢？

三、 資料集介紹與資料集來源

資料筆數共有 892 筆，資料表欄位如下表：

欄位	說明	欄位	說明
PassengerId	乘客編號	SibSp	家屬 (旁系) 同行人數
Survived	是否存活	Parch	家屬 (直系) 同行人數
Pclass	艙等	Ticket	船票號碼
Name	姓名	Fare	票價
Sex	性別	Cabin	客艙號碼
Age	年齡	Embarked	登船地點

資料來源：https://www.kaggle.com/c/titanic/data

四、 資料預處理

由於取得的資料有丟失值，必須預先處理，我們針對下列欄位的缺失值做填補：

- 年齡 Age：由於 Age 有丟失值，先處理丟失值問題。 Age 的丟失值較多，利用隨機森林來填補。
- 票價 Fare：缺失較少，空值直接使用平均值來填補。
- 客艙號碼 Cabin：缺失值改成 no Cabin。
- 登船地點 Embarked：缺失較少，空值直接使用眾數來填補。

再來，我們可以從 Name、SibSp、Parch 欄位中，得知資料的特徵，並將其整合：

- 家庭人數 FamilySize：將家屬 (旁系) 同行人數 Sibsp 和家屬 (直系) 同行人數 Parch 欄位合併在一起。
- 身分 Identity：將姓名中的身分特徵取出。

最後，進行特徵轉換，將 Sex、Embarked、Title、Cabin、Ticket_info 這 5 個，

其中 Sex 屬於二分類，可以用 LabelEncoder 處理，同時原資料刪除。

	Age	Cabin	Embarked	Fare	Name	Parch	PassengerId
0	41.326267	7	1	8.4583	Moran, Mr. James	0	6
1	41.616486	7	2	13.0000	Williams, Mr. Charles Eugene	0	18
2	46.792625	7	0	7.2250	Masselmani, Mrs. Fatima	0	20
3	41.326267	7	0	7.2250	Emir, Mr. Farred Chehab	0	27
4	34.860886	7	1	7.8792	O'Dwyer, Miss. Ellen "Nellie"	0	29

Pclass	Sex	SibSp	Survived	Ticket	Family_Size	Title1	Title2	Ticket_info
2	1	0	0.0	330877	0	12	2	36
1	1	0	1.0	244373	0	12	2	36
2	0	0	1.0	2649	0	13	3	36
2	1	0	0.0	2631	0	12	2	36
2	0	0	1.0	330959	0	9	1	36

五、 機器學習或深度學習方法（使用何種方法）

我們使用隨機森林進行預測分析。詳細請看程式碼。

```
from sklearn.ensemble import RandomForestClassifier

rf = RandomForestClassifier(criterion='gini',
                            n_estimators=1000,
                            min_samples_split=12,
                            min_samples_leaf=1,
                            oob_score=True,
                            random_state=1,
                            n_jobs=-1)

rf.fit(dataTrain.iloc[:, 1:], dataTrain.iloc[:, 0])
print("%.4f" % rf.oob_score_)

pd.concat((pd.DataFrame(dataTrain.iloc[:, 1:].columns, columns = ['variable']),
           pd.DataFrame(rf.feature_importances_, columns = ['importance'])),
          axis = 1).sort_values(by='importance', ascending = False)[:20]

rf_res =  rf.predict(dataTest)
submit['Survived'] = rf_res
submit['Survived'] = submit['Survived'].astype(int)
submit.to_csv('submit.csv', index= False)
```

```
0.8294
```

六、 研究結果及討論（含模型評估與改善）

我們一開始使用迭代決策樹進行特徵選擇，最後以投票法產生結果，但上傳後的結果為：

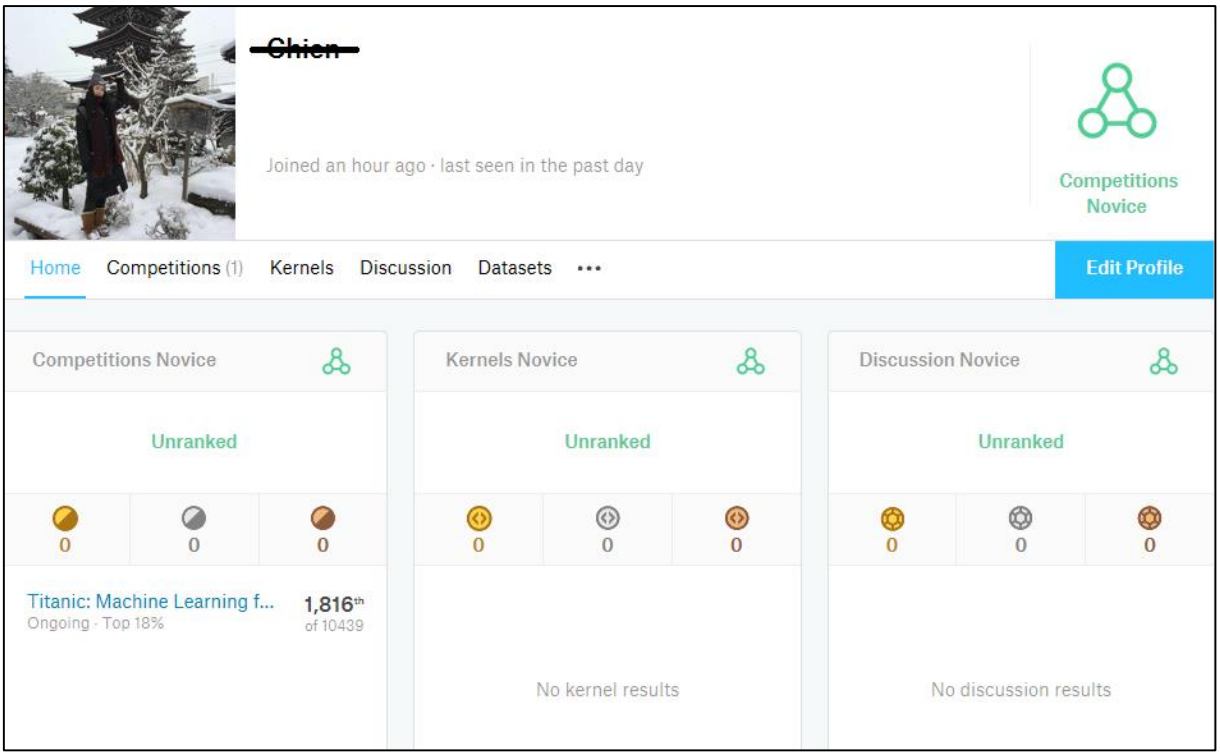
Submission and Description	Public Score	Use for Final Score
submit.csv 12 minutes ago by Chien add submission details	0.79904	☐
result.csv an hour ago by Chien add submission details	0.00000	☐

因成效不彰，所以改為使用隨機森林。

以下為預測結果：

PassengerId	Survived
892	0
893	1
894	0
895	0
896	1
897	0
898	0
899	0
900	1
901	0
902	0
903	0
904	1
905	0
906	1
907	1
908	0
909	0
910	0
911	1
912	0
913	1
914	1
915	0
916	1
917	0
918	1
919	0
920	1
921	0
922	0

Submission and Description	Public Score	Use for Final Score
submit.csv 12 minutes ago by Chien add submission details	0.79904	☐
result.csv an hour ago by Chien add submission details	0.00000	☐



七、 結論

一開始我們嘗試做的結果雖然不如預期，但是我們也從錯誤中學習到了如何使用迭代決策樹及投票法（硬投票）來進行分析與決策，在網路上進一步了解更多機器學習的方法，認知到了隨機森林真是一個好用的方法，讓我們可以很快的解決前面所發生的問題（預測失敗），這也是非常難能可貴的經驗。

八、 參考文獻

- [機器學習專案] Kaggle 競賽-鐵達尼號生存預測(Top 3%)
  https://medium.com/@yulongtsai/https-medium-com-yulongtsai-titanic-top3-8e64741cc11f?fbclid=IwAR1ffhRKTAnzkZreMMKXAwzZdXHW28btblYX0iu_ALzANrSplB0aCoeZzys
- 泰坦尼克號乘客資料分析
  https://zhuanlan.zhihu.com/p/26440212?fbclid=IwAR3KUngpnkMO0uvzuYdF6dq1JVEm6r-wDCi02YbTKVzR3h_wLCmhUCVYkBk
- Titanic 生存預測 1
  https://www.jianshu.com/p/e5b02ba38f3b?fbclid=IwAR1ty_T9ZdbDDV1abBvl0jgHSK8yYPB3vat-wDxnCg1JTgy17aKqyeb7YI0
- Titanic 生存預測 2
  https://www.jianshu.com/p/48c93553e7b6?fbclid=IwAR3YQb4lcEidlyW2pPodEEhXO0FpP5m8w0pc-zvOxtpoKkU8iPLuQ6Are48